

Approximated segmentation considering technical and dosimetric constraints in intensity-modulated radiation therapy with electrons

Antje Kiesel¹ Tobias Gauer²

September 29, 2010

¹ Institute for Mathematics, University of Rostock, Rostock, Germany
E-Mail: antje.kiesel@uni-rostock.de

² Department of Radiotherapy and Radio-Oncology, University Medical Center Hamburg-Eppendorf, Hamburg, Germany
E-Mail: t.gauer@uke.uni-hamburg.de

Abstract

In intensity-modulated radiation therapy, optimal intensity distributions of incoming beams are decomposed into linear combinations of leaf openings of a multileaf collimator (segments). In order to avoid inefficient dose delivery, the decomposition should satisfy a number of dosimetric constraints due to suboptimal dose characteristics of small segments. However, exact decomposition with dosimetric constraints is only in limited cases possible. The present work introduces new heuristic segmentation algorithms for the following optimization problem: Find a segmentation of an approximated matrix using only allowed fields and minimize the approximation error. Finally, the decomposition algorithms were implemented into an optimization programme in order to examine the assumptions of the algorithms for a clinical example. As a result, identical dose distributions with much fewer segments and a significantly smaller number of monitor units could be achieved using dosimetric constraints. Consequently, the dose delivery is more efficient and less time consuming.

Keywords: IMRT planning, intensity matrix, approximated segmentation, dosimetric and technical constraints, multileaf collimator

Mathematics Subject Classification (2000): MSC 90C90, MSC 92C50, MSC 49M25, MSC 49M27

1 Introduction

In intensity-modulated radiation therapy (IMRT), intensity matrices with non-negative integer entries are computed for each irradiation field. After discretization of the field into bixels, each entry of the matrix corresponds to the required intensity within this bixel. The segmentation step consists in decomposing the matrix into a linear combination of subfields (segments) shaped by a multileaf collimator (MLC). The first intuition is that a treatment plan is optimal, if the linear combination of the chosen segments equals the matrix. Such

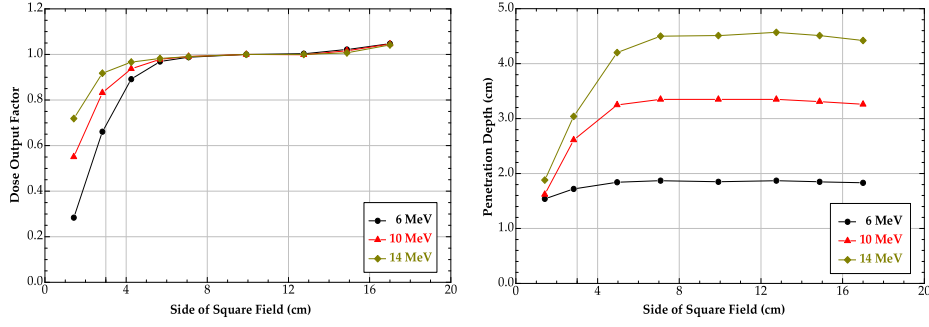


Figure 1: Electron dose output at the dose maximum normalized to the dose output of the 10 cm \times 10 cm field and electron penetration depth of the 90 % depth-dose as a function of square field size and electron energy (from [13]). The fields were shaped by an add-on MLC for electrons presented in Figure 6. A minimum MLC field size of approximately 3 cm \times 3 cm is necessary for decomposing intensity distributions into leaf openings to ensure an output factor of nearly 1 and an energy-dependent penetration depth.

a plan consists of various segments possibly including those segments where most of the irradiation field is covered and only few bixels receive radiation.

For dosimetric reasons, however, the model assumption is not given in practice. Irradiation of small photon or electron segments result in a much lower dose output compared to conventional conformal fields. Therefore, the linearity assumption, that irradiating one segment is equivalent to dividing it into two parts and irradiating them separately, only holds, if the two parts are still sufficiently large. In addition, the penetration depth of electrons decreases with decreasing field size and is almost independent of the beam energy for approximately 1 cm \times 1 cm fields. However, the energy dependence of the penetration depth is necessary for our new IMRT technique with electron beams to adjust the dose to the target volume by use of various beam energies. Figure 1 shows that electron fields of approximately 3 cm \times 3 cm are necessary to keep an output factor of nearly 1 and an energy-dependent penetration depth.

As a consequence, a treatment plan should consist of segment shapes satisfying certain constraints that ensure a minimum field size. For practical purposes it is also necessary that the field openings are connected and do not degenerate into two or more parts. Besides those dosimetric constraints, there are also technical constraints reducing the number of allowed shapes. One is the leaf overtravel constraint that accommodates the fact that the left (respectively right) leaf of the MLC cannot be shifted further than a threshold to the right (respectively left). These constraints have the consequence, that not every intensity matrix is decomposable in segments satisfying the constraints. This leads us to the task to find an approximation matrix and its decomposition into "good" segments, that differs from the given intensity matrix as little as possible. The aim is to generate equivalent treatment plans with good segments leading to a reduction in the segment number and monitor units, respectively.

The decomposition problem for the exact case without concerning any additional constraints is well studied. Algorithms for the minimization of the beam-on time can be found in [2, 3, 6, 14, 15, 21]. Approaches for minimizing the number of used segments are given in [7, 18, 26]. A variety of technical constraints are considered, see [5, 16] for the interleaf collision constraint, that prohibits an overlap of adjacent leaf pairs, and [16, 17, 22, 23, 27] for the

tongue-and-groove constraint. Kamath et. al. [21] also investigate the minimum separation constraint that requires a minimum leaf opening in each row and develop a criterion for a matrix being decomposable under this constraint. Engelbeen and Fiorini [10] deal with the interleaf distance constraint where the allowed difference between two left (respectively right) leaf positions is bounded by some given threshold.

An approximation problem with the aim of reducing the total beam-on time was first formulated in [8] and generalized to approximated decomposition with interleaf-collision constraint in [19] and [20]. The dependence between field size and output factors, penetration depth and depth-dose fall-off is outlined in [13]. These considerations lead to the decomposition problem using segments that satisfy some minimum field size constraints. Under these constraints, an exact decomposition of the intensity matrices is, in general, no longer possible (cf. [21]) and an approximation problem has to be formulated.

Another algorithmic approach that aims at minimizing the number of segments while keeping the quality of the treatment plan is the direct aperture optimization that combines the choice of beams, apertures and weights without computing a leaf sequencing step. Shepard et al. [28] allow only a limited number of apertures for each beam, Bedford and Webb [4] also integrate constraints on the segment shape and size in the direct aperture optimization approach. Our algorithm is applicable if one uses intensity profile segmentation and wants to compute segmentations satisfying certain field size constraints and reducing the complexity of the plan. Matuszak et al. [25] deal with the minimization of the monitor units by smoothing the intensity profiles.

The paper is organized as follows. Section 2 gives two definitions of what we call a segment with good dosimetric properties, one basic definition and an extended one including one further constraint. We concretely define the approximation problem and in Section 3 propose heuristic algorithms for both definitions, each of them consisting of seven different steps. The different parts of the algorithm and their properties are analyzed in Section 4. We especially outline, that the solutions of the subproblems in step 1 and 2 are indeed optimal. Section 5 introduces the clinical case we used for testing the quality of our segmentations. Section 6 gives computational results for the test case and detailed numerical results for the segmentation of clinical matrices from different IMRT treatment plans.

2 Problem formulation and definitions

Throughout the paper we use the notation

$$[k] = \{1, 2, \dots, k\} \quad \text{and} \quad [k, l] = \{k, k+1, \dots, l\}$$

for integers k and l , $k \leq l$. Let $A = (a_{ij})$ denote the given fluence matrix of size $m \times n$. Feasible leaf positions of the MLC are modeled as binary matrices $S = (s_{ij})$, called *segments*, that satisfy the consecutive-ones-property in each row. In other words, S is a segment, if there are integral intervals $[\ell_i, r_i]$ for all $i \in [m]$, representing the positions of the left and the right leaf, such that

$$s_{ij} = \begin{cases} 1 & \text{if } \ell_i \leq j \leq r_i \\ 0 & \text{otherwise} \end{cases} \quad ((i, j) \in [m] \times [n]). \quad (1)$$

Furthermore, for each segment S , we define $s_{0,j} = s_{m+1,j} = 0$ for all $j \in [n]$.

For the described reasons, we introduce five parameters b_l , b_r , g_1 , g_2 and f representing the following constraints:

- (i) Left Leaf Overtravel Constraint: For all $i \in [m]$, we require $l_i \leq b_l$. In each row the left leaf cannot be shifted more to the right than to the bixel with index b_l .
- (ii) Right Leaf Overtravel Constraint: For all $i \in [m]$, we require $r_i \geq b_r$. In each row the right leaf cannot be shifted more to the left than to the bixel with index b_r .
- (iii) Minimum Separation Constraint and Row Overlap: If a row $i \in [m]$ is not totally covered, we require $r_i - l_i \geq g_1 - 1$. Similarly, if rows i and $i + 1$ are not completely covered, we claim $\min(r_i, r_{i+1}) - \max(l_i, l_{i+1}) \geq g_1 - 1$. At least g_1 consecutive bixels in each row receive radiation and the irradiated area of two consecutive rows overlaps in at least g_1 bixels.
- (iv) Minimum Vertical Gap: We require a minimum vertical field size and a minimum vertical size of the covered regions, i.e. in each column consecutive ones or zeros should have a minimum number g_2 . In detail, if $s_{i-1,j} = 0$, $s_{ij} = s_{i+1,j} = \dots = s_{k-1,j} = 1$ and $s_{kj} = 0$ for some column j , we have $k - i \geq g_2$. Analogously, we require the same for consecutive zeros framed by ones.
- (v) Minimum Total Field Height: At least f consecutive rows of the field are not totally covered, i.e. there are at least f consecutive rows with $l \leq r$. This ensures, that the total size of the field is reasonably large.

Of course, these parameters only make sense if $1 \leq b_l, b_r, g_1 \leq n$, $1 \leq g_2, f \leq m$ as well as $1 \leq b_r < b_l \leq n$ and $g_2 \leq f$.

The case that one row of the field is totally covered and receives no radiation at all, is throughout this paper represented by the leaf positions $l = n + 1$ and $r = 0$. In practice, one will of course choose leaf positions of the form $l = r + 1$ with $l \leq b_l$ and $r \geq b_r$ that respect the leaf overtravel constraints (i) and (ii).

Remark 1. The Minimum Vertical Gap can be formulated in terms of the leaf position as follows: If $l_i < l_{i-1}$ for some $i \geq 2$, then we also require $l_{i+1} \leq l_i, l_{i+2} \leq l_i, \dots, l_{i+g_2-1} \leq l_i$. Analogously, if $r_i > r_{i-1}$ for some $i \geq 2$, we also have $r_{i+1} \geq r_i, r_{i+2} \geq r_i, \dots, r_{i+g_2-1} \geq r_i$. This ensures, that in vertical direction, we always have at least g_2 bixels open. Therefore, we additionally not allow that $l_i > l_{i-1}$ or $r_i < r_{i-1}$ for $2 \leq i \leq g_2$ and forbid also $l_i < l_{i-1}$ as well as $r_i > r_{i-1}$ for $m - g_2 + 1 \leq i \leq m$. Similarly, we require at least g_2 bixels closed in vertical direction, if there are open bixels above and below in this column of the matrix. Thus, we make sure that also thin shapes in vertical direction, having negative dosimetric properties as discussed in the introduction, are forbidden.

A segment is called *connected* if the irradiated area that corresponds to its leaf positions does not resolve into two or more parts, i.e. if the corresponding rectilinear polygon (considered as an open set) is connected.

As the realization of the minimum vertical gap turns out to be the most difficult task, we introduce two different definitions of “good” segments.

Definition 1. Given the parameters $b_l, b_r, g_1, f \in \mathbb{N}$ with $1 \leq b_l, b_r, g_1 \leq n$, $1 \leq f \leq m$ as well as $1 \leq b_r < b_l \leq n$, a *segment with good dosimetric properties* S is a connected segment satisfying the constraints (i), (ii), (iii) and (v).

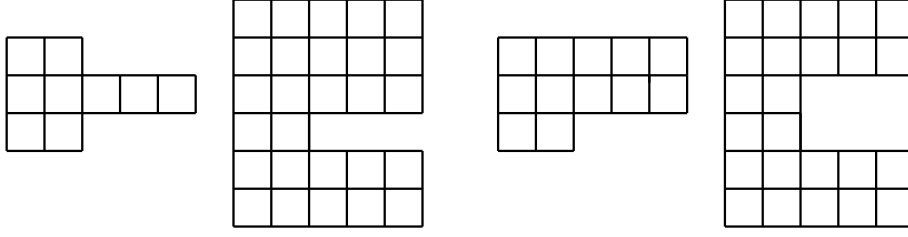


Figure 2: For $g_1 = g_2 = 2$ the two left segments violate the minimum vertical gap constraint and only satisfy (iii), whereas the two right segments meet both conditions (iii) and (iv).

Definition 2. Given the parameters $b_l, b_r, g_1, g_2, f \in \mathbb{N}$ with $1 \leq b_l, b_r, g_1 \leq n$, $1 \leq g_2, f \leq m$ as well as $1 \leq b_r < b_l \leq n$ and $g_2 \leq f$, a *segment with very good dosimetric properties* is a connected segment satisfying the constraints (i)-(v).

For brevity of notation we will call the segments with good dosimetric properties from now on simply segments and the segments with very good dosimetric properties advanced segments.

All in all, we have two optimization problems: Given a matrix A with positive integer entries and suitable parameters b_l, b_r, g_1, g_2, f , find a segmentation

$$B = \sum_{k=1}^t u_k S_k$$

where the u_k are positive integers and

- **Approximated Segmentation into segments (ASS):** the S_k are segments
- **Approximated Segmentation into advanced segments (ASAS):** the S_k are advanced segments

such that

$$\sum_{i=1}^m \sum_{j=1}^n |a_{ij} - b_{ij}| \rightarrow \min.$$

The value of the objective function of the optimization problem is called *total change*. The delivery time of the segmentation is $\sum_{k=1}^t u_k$ and the number of segments is t .

Obviously, for the parameter set $b_l = n, b_r = 1, g_1 = g_2 = f = 1$, segments are simply connected segments in the sense of Equation (1), the approximation problem has 0 as value of the objective function and degenerates to a segmentation problem into connected segments defined by Equation (1).

3 Approximated segmentation

Now we introduce two basic algorithms for **ASS** and **ASAS**, each consisting of seven different steps. Two of the steps are identical in both algorithms as they are computed before the segmentation step and therefore do not affect the parameter g_2 . Five of the steps differ subject to whether the constraint g_2

is regarded or not (namely steps 3-7). Steps 1 and 2 will be solved exactly, whereas steps 3-7 are heuristic.

The basic structure of our algorithms is as follows:

1. Solve the **Leaf Overtravel Constraint Problem (LOC)** on A : Given $b_l, b_r \in \mathbb{N}$ with $1 \leq b_l < b_r \leq n$, find an approximation B with nonnegative integer entries that can be decomposed with respect to the Leaf Overtravel Constraint such that the total change is minimal.
 $A := B$.
2. Solve the **Minimum Separation Constraint Problem (MSC)** on A : Given $g_1 \in [n]$, find an approximation B that can be decomposed into segments with $r_i - l_i \geq g_1 - 1$ for all $i \in [m]$ such that the total change is minimal.
 $A := B$.
3. **ASS**: Compute an approximated segmentation \mathcal{S} into connected segments satisfying (i)-(iii), but not necessarily (v).
ASAS: Compute an approximated segmentation \mathcal{S} into connected segments satisfying (i)-(iv), but not necessarily (v).
 The approximated segmentation may violate (v) and belongs to an approximation matrix B that might have a large total change.
4. **Combination of Fields**: Combine stepwise two disjoint fields, if this is possible with small total change. For ASAS, make sure that the new field satisfies (iv).
5. **Make-two-of-two**: For $S, S' \in \mathcal{S}$, where S violates (v) and S' satisfies (v), compute a substitution $S + S' = \hat{S} + \hat{S}'$, such that \hat{S} and \hat{S}' satisfy (v). For ASS, make sure that \hat{S} and \hat{S}' satisfy (i)-(iii). For ASAS, \hat{S} and \hat{S}' must also satisfy (iv).
6. **Handle Critical Segments**: If there are still segments violating (v), try to combine them with a feasible segment such that the total change of the combination is smaller than omitting the critical segment. Take care, that (i)-(iii) (resp. (i)-(iv)) hold.
7. **Total Change Improvement**: For all segments and all rows i , check whether an increase or decrease of l_i (respectively r_i) reduces the total change. For ASS, change l_i or r_i only if (i)-(iii) still hold. For ASAS, change l_i or r_i only if (i)-(iv) still hold. Look at all the segments cyclically until no more changes are possible.

The output of the algorithm is a segmentation of an approximation matrix that consists only of segments for ASS and only of advanced segments for ASAS. The LOC- and the MSC-approximation aim at producing a first approximation that can be better decomposed with the given constraints (i)-(iii) in the segmentation step than the initial matrix. The combination of fields, the make-two-of-two-step and the handling of critical segments try to provide segments satisfying (v) without producing too much total change. Finally, the total change improvement is computed in order to improve the approximation.

A very important feature of our heuristic is that further constraints can easily be taken into consideration. For example, if adjacent left and right leaves may not overlap (interleaf collision constraint) one can allow only leaf positions that respect this constraint in the optimization steps 3-7.

4 Subproblems

Now we describe in detail the subalgorithms and outline some of their basic properties. Throughout the steps that follow the leaf overtravel approximation, we allow only leaf positions that respect the leaf overtravel constraint. For simplicity, we will not mention this basic fact in every step.

4.1 Leaf Overtravel Constraint

As the leaf overtravel constraint only affects a single row of the matrix, the problem LOC can be solved for each row independently. Thus, we compute an optimal approximation of a vector \mathbf{a} . Segmentations reduce to sums of intervals $[l, r]$. Segments are simply 0-1-vectors \mathbf{s} with consecutive ones.

Lemma 1. *A vector \mathbf{a} has a segmentation $\mathbf{a} = \sum_{i=1}^k \mathbf{s}_i$ with corresponding leaf positions $l_i \leq b_l$ and $r_i \geq b_r$ iff $a_j \geq a_{j+1}$ for all $j \in [b_l, n-1]$ and $a_j \geq a_{j-1}$ for all $j \in [2, b_r]$.*

Proof. Let $a_0 = a_{n+1} := 0$ and $a_+ := \max\{0, a\}$. On the one hand, the algorithm of Bortfeld (see [6]) provides a segmentation where the left leaf position is j for exactly $(a_j - a_{j-1})_+$ segments. Analogously, the right leaf position is j for $(a_j - a_{j+1})_+$ segments and no other leaf positions occur. On the other hand, it is obvious that if $a_j > a_{j-1}$ (respectively $a_j > a_{j+1}$) there will be a segment with left (respectively right) leaf position j in every segmentation. This concludes the proof. \square

Therefore, we have to find an approximation vector, that has no up-steps after index b_l and no down-steps before index b_r . As we assume $b_r < b_l$, we can use symmetry to solve the approximation problem for the right leaf positions. Besides, the criterion from Lemma 1 shows, that $b_j = a_j$ for $j \in [b_r+1, b_l-1]$ for each optimal solution of LOC. We simply need to solve the following problem for the subvector (a_{b_l}, \dots, a_n) and the left overtravel constraint:

LOC-left: Given a vector $\mathbf{v} = (v_1, \dots, v_k)$, find an approximation vector \mathbf{w} with $w_j \geq w_{j+1}$ for $j \in [k-1]$ such that $\|\mathbf{v} - \mathbf{w}\|_1 = \sum_{j=1}^k |v_j - w_j| \rightarrow \min$.

The algorithm for solving the problem LOC-left is described in Algorithm 1 in the appendix. It uses a graph theoretical approach and computes a shortest path in a layered digraph, where the j -th layer consists of nodes representing the possible entries of the j -th component of the approximation vector. The problem LOC-left is similar to the Monotone Discrete Approximation Problem (MDAP) formulated in [8] and the algorithm follows the same idea.

Let $\min := \min_{j \in [k]} v_j$ and $\max := \max_{j \in [k]} v_j$ and let tc_{ij} be the objective value of an optimal solution of LOC-left with $w_j = i$. Let pre_{ij} be the corresponding predecessor w_{j-1} . With respect to Algorithm 1 (that uses the notation above) we yield the following

Theorem 1. *Algorithm 1 computes an optimal solution of LOC-left.*

Proof. The initial values tc_{i1} are trivially correct. Let now $j > 1$ and let (w_1, \dots, w_j) be an optimal approximation of (v_1, \dots, v_j) with $w_j = i$. By induction, $tc_{w_{j-1}, j-1}$ is computed correctly and thus

$$\sum_{l=1}^j |v_l - w_l| = tc_{w_{j-1}, j-1} + |v_j - i| \geq tc_{ij}.$$

Therefore tc_{ij} is a lower bound for the total change. The choice of i_{opt} makes sure, that the optimal value of w_k is chosen and obviously, the approximation vector from Algorithm 1 realizes the lower bound for the total change of $tc_{i_{opt},k}$. \square

4.2 Minimum separation constraint

Like the Overtravel Constraint, the Minimum Separation Constraint can be handled independently for each row of the matrix. Thus the task is the following:

MSC-Row Given a vector $\mathbf{v} = (v_1, \dots, v_n)$ with nonnegative integer entries, find an approximation vector \mathbf{w} with nonnegative integer entries, such that \mathbf{w} has a decomposition into intervals of length $\geq g_1$ and $\|\mathbf{v} - \mathbf{w}\|_1 = \sum_{j=1}^n |v_j - w_j| \rightarrow \min$.

We know from Kamath et al. (see [21]) that a vector \mathbf{a} can be decomposed without violating the minimum separation constraint, if the optimal decomposition of their algorithm SINGLEPAIR does not violate the minimum separation constraint. For example, the vector $\mathbf{a} = (1, 2, 1)$ cannot be decomposed with $g_1 = 3$, as the optimal decomposition is $(1, 2, 1) = (1, 1, 0) + (0, 1, 1)$ and the used intervals do not have a minimum length of 3. This motivates the approximation problem defined above.

Obviously, the problem MSC-Row can be formulated as an integer linear programming problem as follows:

$$\begin{aligned}
\sum_{j=1}^k \sum_{j'=\min(k,j+g_1-1)}^n u_{j,j'} - \gamma_k &\leq a_k & k \in [n] \\
-\sum_{j=1}^k \sum_{j'=\min(k,j+g_1-1)}^n u_{j,j'} - \gamma_k &\leq -a_k & k \in [n] \\
u_{j,j'} &\geq 0 & j, j' \in [n], j' \geq j + g_1 - 1 \\
u_{j,j'} &\in \mathbb{Z} & j, j' \in [n], j' \geq j + g_1 - 1 \\
\gamma_k &\in \mathbb{Z} & k \in [n] \\
\sum_{k=1}^n \gamma_k &\rightarrow \min
\end{aligned}$$

We solve this integer program for each row of the matrix using SCIP [1] with SoPlex [29] as LP solver. The problem can also be solved by a combinatorial algorithm using a minimum cost flow formulation which is shown in [11].

4.3 Segmentation

Let B be the approximation matrix resulting from the MSC-step. The basic idea of the segmentation is to consider the current total change to be the sum of the absolute values of the entries of B and to iteratively compute a segment S whose subtraction reduces the current total change. In each step, the matrix B is updated by setting $B := B - S$. At the end of the segmentation, a positive entry in B represents a bixel with underdose and a negative entry a bixel with overdose.

4.3.1 Segmentation for ASS

Let b_i denote the i -th row of B for all $i \in [m]$. A segmentation consists of segments S each represented by its leaf positions l_i, r_i for $i \in [m]$. The main body for the segmentation step is described in Algorithm 2 in the appendix. This algorithm uses the subroutine **Find interval ASS** that is precisely described in Algorithm 3.

The idea behind this heuristic choice of the segment S being subtracted from B in each step is, that we compute the first interval from a sliding window segmentation (see again [6]), with l as index of the first up-step and r as index of the first down-step in the corresponding row. If these values already satisfy all requirements, we stop. Otherwise, we lengthen the interval by changing B , such that the overlap with the previous row increases. In order to keep the total change small, we neglect this approximation and close the row, if there is no overlap at all.

The segmentation resulting from Algorithm 2 satisfies the constraints (i)-(iii) and the connectedness, but may contain segments that do not have the minimum total field height f .

4.3.2 Segmentation for ASAS

The segmentation step for ASAS differs a little bit from the ASS segmentation. For ASS, we always find a segment that has its first nonzero row exactly in that row where the current matrix B has its first nonzero row. Going through the rows, we add further ones to the segment if the current sliding window interval overlaps with the previous row.

Computational tests have shown that for ASAS a different technique makes sense because the vertical criterion (iv) plays a role. In detail, whenever we decide for $l_{i+1} < l_i$ (respectively $r_{i+1} > r_i$), this immediately implies $l_{i+k} < l_i$ (respectively $r_{i+k} > r_i$) for $k = 2, \dots, g_2$. Additionally, as we need at least g_1 consecutive ones in each row, we also know $r_{i+k} \geq \max(l_i - 1, l_{i+1} + g_1 - 1)$ (respectively $l_{i+k} \leq \min(r_i + 1, r_{i+1} - g_1 + 1)$). Therefore, we use a matrix $S = (s_{ij})$ to store unavoidable ones, i.e. if $l_{i+1} < l_i$, we put

$$s_{i+k,j} = 1 \text{ for } 2 \leq k \leq g_2, l_{i+1} \leq j \leq \max(l_i - 1, l_{i+1} + g_1 - 1) \quad (2)$$

and if $r_{i+1} > r_i$, we define

$$s_{i+k,j} = 1 \text{ for } 2 \leq k \leq g_2, \min(r_i + 1, r_{i+1} - g_1 + 1) \leq j \leq r_{i+1} \quad (3)$$

As we require consecutive ones in each row, we also put

$$s_{i+k,j} = 1 \text{ for } 2 \leq k \leq g_2, l_{i+1} \leq j \leq r_{i+1} \quad (4)$$

if $l_{i+1} < l_i$ and $r_{i+1} > r_i$ at the same time.

We also have to take care that the covered regions have a vertical minimum size. If $l_{i+1} > l_i$ (respectively $r_{i+1} < r_i$), we analogously put unavoidable zeros into our matrix S using the corresponding rules to (2) and (3). (4) is not necessary here, as zeros do not have to be consecutive in the rows.

Example 1. Let $g_1 = g_2 = 3$ and let the $(*)$ -entries of the matrix above denote the open bixels for row 1 and 2. Before choosing the leaf positions for

row 3, there are some unavoidable ones and zeros that have to be respected.

$$\begin{pmatrix} & * & * & * & * \\ * & * & * & * & \\ 1 & 1 & 1 & & 0 \\ 1 & 1 & 1 & & 0 \end{pmatrix}$$

Thus, the choice of the leaf positions in one row produces unavoidable ones or zeros in other rows. Our algorithm will choose l_i and r_i such that the total change of this row and the corresponding unavoidable ones is minimal. Therefore it might happen, that we do not use the first nonzero row and we also do not use the sliding window technique anymore, as the minimum vertical gap constraint (iv) prohibits so many leaf positions that we can compare the remaining ones with regard to the resulting total change. Algorithm 4 and 5 in the appendix show the corresponding segmentation steps.

The idea behind this algorithm is, that for each segment and in each row we look at all feasible leaf positions. For each pair (l, r) , we compute the value *benchmark*, which is the difference between the number of positive entries and the number of nonpositive entries in this row as well as in the corresponding unavoidable ones. The larger this value is, the better the pair (l, r) suits to the segmentation. The unavoidable zeros are not taken into account because it is not necessarily bad if an entry $b_{i,j} > 0$ is closed, as this entry can be part of the following segments. We close a row, if the corresponding optimal value of benchmark is zero.

Finally, our procedure leads to a segmentation satisfying (i)-(iv) and the connectedness, but not necessarily satisfies the minimum total field height f .

Remark 4. One might argue that it is possible for ASS to compute exactly the segment, that reduces the total change in this step as much as possible. For example, one can consider a layered digraph with m layers of nodes. In layer $i \in [m]$, we have nodes (i, l, r) representing feasible leaf positions and we draw an edge between (i, l, r) and $(i + 1, l', r')$, if the combination of these two leaf positions satisfies all constraints. Furthermore, we draw edges from a source to all nodes in the first layer and from all nodes in the last layer to a sink. The edge weights are just the total change reductions caused by the leaf positions of the end node of the edge and zero for all edges whose end node is the sink. For a detailed description of the graph see [5]. The optimal segment can then be found by shortest path computation in the digraph. But indeed, such a choice is not a good idea because reducing the total change as much as possible leads to badly decomposable residual matrices. For example, for $g_1 = 2$, $\mathbf{b} = (1, 2, 1)$ would be reduced by $(1, 1, 1)$ and the residuum $(0, 1, 0)$ is badly decomposable. Our used sliding-window-technique is better and leads to $(1, 2, 1) = (1, 1, 0) + (0, 1, 1)$.

For ASAS the constraints are too complex anyway to compute the optimal segment in one step, that reduces the total change by a maximum value.

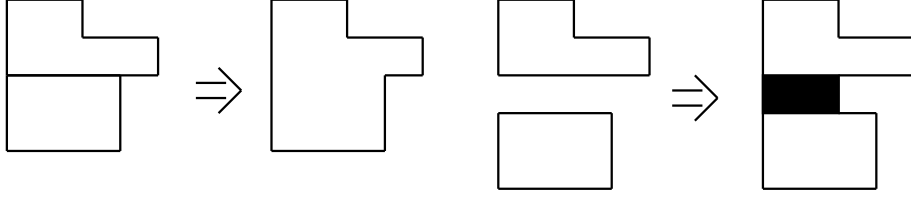


Figure 3: The first and the second combination step for ASS with $g_1 = 2$. The black rectangle produces an increase of the total change.

4.4 Combination of fields

Given two segments $S = ((l_1, r_1), \dots, (l_m, r_m))$ and $S' = ((l'_1, r'_1), \dots, (l'_m, r'_m))$, let us consider the open regions i_1, \dots, i_2 and i'_1, \dots, i'_2 of S and S' , precisely

$$\{i_1, \dots, i_2\} = \{i \in [m] : l_i < n + 1\} \quad (5)$$

$$\{i'_1, \dots, i'_2\} = \{i \in [m] : l'_i < n + 1\} \quad (6)$$

If $i'_1 = i_2 + 1$ and $\min(r_{i_2}, r'_{i'_1}) - \max(l_{i_2}, l'_{i'_1}) \geq g_1 - 1$, we merge S and S' and get one new segment S'' with

$$s''_{ij} = \begin{cases} 1, & \text{if } s_{ij} = 1 \text{ or } s'_{ij} = 1, \\ 0, & \text{otherwise.} \end{cases}$$

We iterate this step, until no two segments can be merged by this procedure. Obviously, this step does not affect the total change.

Afterwards, we compute a second combination step and merge segments if $i'_1 = i_2 + 2$ and $\min(r_{i_2}, r'_{i'_1}) - \max(l_{i_2}, l'_{i'_1}) \geq g_1 - 1$. This means, there is only one closed row between the two segments. We compute leaf positions $l \in [\max(l_{i_2}, l'_{i'_1}), \min(r_{i_2}, r'_{i'_1}) - g_1 + 1]$ and $r \in [l + g_1 - 1, \min(r_{i_2}, r'_{i'_1})]$ such that putting ones to the interval $[l, r]$ in row $i_2 + 1$ produces the smallest total change with respect to the current approximation matrix.

Again, we drop S and S' out of our segmentation and this time add S'' with

$$s''_{ij} = \begin{cases} 1, & \text{if } s_{ij} = 1 \text{ or } s'_{ij} = 1 \text{ or} \\ & (i = i_2 + 1, l \leq j \leq r), \\ 0, & \text{otherwise.} \end{cases}$$

Again, we iterate this procedure, until no more such merges are possible. Obviously, this second combination step affects the total change, as we increase the approximation matrix by adding ones to the segments. After both of our combination steps, all segments still satisfy (i)-(iii), while even more segments satisfy (v) now. Another positive consequence is a reduction of the total number of (not necessarily pairwise different) segments, called the Delivery Time. The combination step is demonstrated in Figure 3.

For ASAS, we only combine two segments according to one of the two steps described above, if the criterion with the minimum vertical zeros (iv) is not violated after the combination.

4.5 Make-two-of-two

We now define a substitution step $S + S' = \hat{S} + \hat{S}'$. For ASAS, the same step is computed if \hat{S} and \hat{S}' still satisfy (iv).

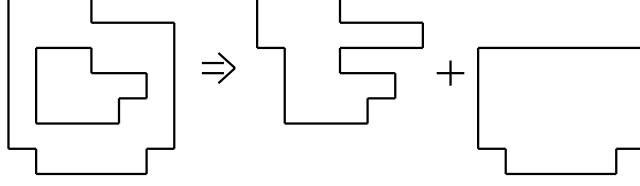


Figure 4: The Make-two-of-two-step for ASS with $g_1 = 2$ and $f = 4$.

Let us again consider a segment $S = ((l_1, r_1), \dots, (l_m, r_m))$ and its open region i_1, \dots, i_2 defined by (5). If S violates (v), we call S a critical segment. Now we check whether we find a segment $S' = ((l'_1, r'_1), \dots, (l'_m, r'_m))$ with open region i'_1, \dots, i'_2 defined by (6) such that

$$i_2 - i'_1 \geq f - 1 \text{ and } i'_2 - i_1 \geq f - 1.$$

Note that in this case, we have $i'_1 < i_1$ and $i'_2 > i_2$ due to $i_2 - i_1 < f - 1$. Thus, the set of open rows of S is a subset of the set of open rows of S' . If

$$\min(r_{i_1}, r'_{i_1-1}) - \max(l_{i_1}, l'_{i_1-1}) \geq g_1 - 1, \quad (7)$$

we substitute S and S' by segments $\hat{S} = ((\hat{l}_1, \hat{r}_1), \dots, (\hat{l}_m, \hat{r}_m))$ and $\hat{S}' = ((\hat{l}'_1, \hat{r}'_1), \dots, (\hat{l}'_m, \hat{r}'_m))$ defined as follows

$$(\hat{l}_i, \hat{r}_i) = \begin{cases} (l'_i, r'_i) & \text{if } i < i_1, \\ (l_i, r_i) & \text{if } i \geq i_1, \end{cases} \quad (\hat{l}'_i, \hat{r}'_i) = \begin{cases} (l_i, r_i) = (n+1, 0) & \text{if } i < i_1, \\ (l'_i, r'_i) & \text{if } i \geq i_1. \end{cases}$$

The result is that we add the upper part of segment S' to segment S in order to enlarge S , while S' remains sufficiently large. If (7) is not satisfied, we can have a second try and check whether

$$\min(r_{i_2}, r'_{i_2+1}) - \max(l_{i_2}, l'_{i_2+1}) \geq g_1 - 1. \quad (8)$$

If this condition is true, we can analogously add the lower part of S' to S and close all rows $\geq i_2 + 1$ of S' . The Make-two-of-two-procedure is illustrated in Figure 4.

The make-two-of-two-step is computed for all critical segments. If we find a partner, we compute the substitution immediately. If no partner is found throughout the segmentation, the segment S is dropped and stored in a new list of critical segments violating (v). Whereas the substitution with a partner does not affect the total change, the elimination of segments from the segmentation that find no partner leads to an increase of the total change.

4.6 Handle Critical Segments

Let $S = ((l_1, r_1), \dots, (l_m, r_m))$ be a critical segment stored in the make-two-of-two-step and let i_1, \dots, i_2 be its open region defined by (5). We try to combine S with a partner $S' = ((l'_1, r'_1), \dots, (l'_m, r'_m))$ from the segmentation with open region i'_1, \dots, i'_2 defined by (6) if this combination step causes lower total change then simply omit S .

More precisely, if $i'_2 \geq i_2$ and $i'_1 \leq i_1$, we define a new segment $S'' = ((l''_1, r''_1), \dots, (l''_m, r''_m))$ with

$$(l''_i, r''_i) = \begin{cases} (\min(l_i, l'_i), \max(r_i, r'_i)), & \text{if } i_1 \leq i \leq i_2, \\ (l'_i, r'_i), & \text{otherwise.} \end{cases}$$

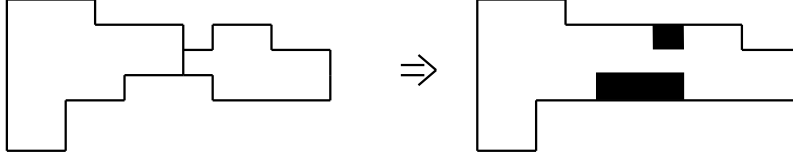


Figure 5: The Handle-critical-segments-step for ASS with $g_1 = 2$ and $f = 4$. The black rectangle shows the combining bixels increasing the total change.

The resulting segment S'' emanates from S' by attaching certain (or all) parts of S and possibly also adding connecting ones. We now drop S' from our segmentation and S from the list of critical segments and add S'' to the segmentation, if the total change that is caused by this decision is smaller than simply omitting S . If no partner for S is found, we simply delete S from the list of critical segments and accept the corresponding total change.

For ASAS, we execute the step if (iv) is not violated.

4.7 Total Change Improvement

As the previous segmentation steps are indeed not optimal, it may happen, that some leaf positions can be increased (respectively decreased) without violating the constraints and the total change becomes smaller.

Let B be the approximation matrix that corresponds to the current segmentation. The improvement step is computed using Algorithm 6 which is described in the appendix and adjusts the segmentation such that B gets closer to A with regard to the total change.

5 Clinical case

A clinical case was set-up to examine the efficiency of our proposed segmentation algorithms. For a patient with cancer of the right breast, electron irradiation plans using various segmentation settings and different optimization constraints were created with a self-designed IMRT optimization programme based on our previous studies [9]. The planning target volume was the right breast, which should receive a total dose of 50.4 Gy (1.8 Gy per fraction). In addition, the target volume should be covered by the 95% isodose line (95% of the prescribed dose). The ipsilateral lung was considered to be organ at risk.

The optimization programme provides simultaneous optimization of beam orientation, energy and intensity for dose delivery with an add-on MLC for electrons (Euromechanics, Schwarzenbruck, Germany) presented in Figure 6 and [12, 13]. Electron dose calculation was performed by Monte Carlo simulations with the treatment planning system *Pinnacle* from Philips (Version 8.1s). Final dose calculation of the treatment plans was conducted using a dose grid size of 3 mm and a dose calculation uncertainty of 2%. The following optimization steps are necessary to generate an electron IMRT plan:

1. Simultaneous optimization of beam orientation, energy and intensity: A set of radiation incidence angles (typically 10–15) is determined given by table and gantry angles [9, 24]. For each configuration, the algorithm calculates the optimal fluence distribution, given by a nonnegative integer matrix.

2. The intensity matrices are approximately decomposed into a superposition of allowed segments such that the deviation between desired and actual fluence is minimal. The result is a set of segments, each of them is given by the corresponding MLC leaf positions and its dose weight.
3. The segments from step 2 are treated as candidates for the treatment plan. In a third optimization step, the dose of the candidates are calculated for all beam energies and then optimized for a given weight proportion between best target coverage and minimum dose to critical organs in order to find the final set of segments with optimal beam energies and their corresponding monitor units.

In this paper, we have focused on step 2 and introduced optimization algorithms for an approximate decomposition of intensity matrices.

Until now, the segmentation step consisted in exactly decomposing the intensity matrices using all deliverable segments. In our approach, we admit a decrease in the decomposition accuracy in order to obtain segments which satisfy the dosimetric and technical constraints. Step 3 justifies the approximation approach in Step 2, as a larger approximation error does not necessarily result in a suboptimal treatment plan. Indeed, larger segments produce homogeneous dose distributions and thus, the same final fluence can be generated using fewer larger segments. The acceptability of a treatment plan is decided after step 3 by means of dose volume histograms (see Section 6) and a plan is only presumed if the required dose constraints are not exceeded. Therefore, the danger of cumulative deviation in the approximation step does not really exist, as the computed segments are just candidates for the treatment plan that pass through a further optimization step.

6 Results

At first, we compare electron IMRT plans created with different segmentation settings for the clinical case prescribed in Section 5. The comparison was conducted using two different optimization settings: one setting to achieve with a better dose coverage of the breast and the other one to reach a better sparing of the lung. Note, that the segmentation settings refer to parameters in the segmentation step, which is discussed here, whereas the optimization settings play a role in the final optimization step that is not part of this paper. Finally, we give a detailed evaluation for the results of the decomposition step.

A treatment plan with a segmentation setting xyz uses the decomposition algorithm with a minimum total field height of $f = x$, a minimum separation constraint and row overlap of $g_1 = y$ and a minimum vertical gap of $g_2 = z$. The decomposed matrices vary in their vertical size m and their horizontal size n , as they describe only parts of the beam head where the target volume is located. Thus, in practice, the overtravel parameters b_l and b_r will depend on the positioning of the matrix and are put individually for each matrix. Our electron MLC is capable of shifting the leaves edges to $3/4$ of the radiation field.

The plan quality was evaluated by means of dose volume histograms that indicate the amount of dose delivered to a certain volume of the patient (here: the right breast and the right lung). Thus, dose homogeneity in the target volume and dose exposure to the organs at risk can be examined. In Figure 7, the dose volume histograms for both optimization settings demonstrate that almost identical dose distributions can be achieved using smaller or larger mini-

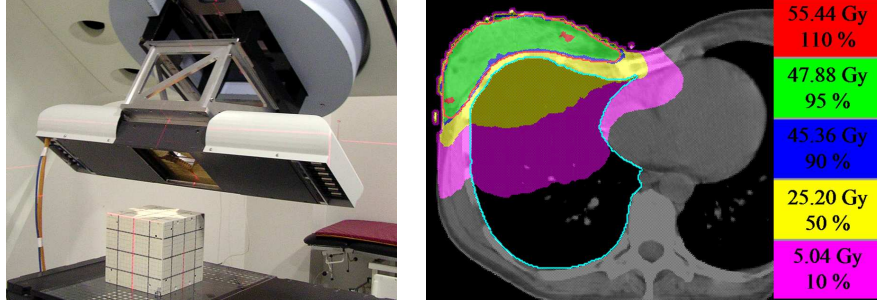


Figure 6: Left: Add-on MLC for electrons mounted on a conventional Siemens accelerator. Right: Dose distribution of an electron IMRT plan consisting of 26 MLC fields achieved through segmentation setting 442. The corresponding dose volume histogram is shown in Figure 7 (left). The setting 442 is given by a minimum total field height of 4, a minimum separation constraint and row overlap of 4 and a minimum vertical gap of 2.

mum MLC openings (cf. setting 111 and 441). In fact, the treatment plan could be slightly improved by use of a minimum vertical gap parameter of 2 which avoids single leaf openings and closings.

Table 1 and 2 illustrate the main benefit of our approach, as identical results can be achieved with approximately two thirds fewer segments and a significantly smaller number of monitor units by use of dosimetric constraints. As a result, the dose delivery is more efficient and less time consuming. The minimum number of segments is reached for setting 442 and computational tests have shown, that this configuration produces the optimal results. As the leaf width is 0.7 cm, fields with a horizontal and vertical height of 4 bixels have a size of approximately $3 \text{ cm} \times 3 \text{ cm}$ and this confirms our dosimetric constraint of $3 \text{ cm} \times 3 \text{ cm}$ minimum segment size (cf. Figure 1). It can be also demonstrated that minimum segment sizes greater than setting 442 do not necessarily result in fewer segments (cf. Table 1), although the number of segments in Table 2 is slightly lower for setting 552. For both optimization settings, the dose volume histograms were considerably better when using minimum segment sizes smaller than setting 552.

It is important to underline that the number of segments and the number of monitor units in Table 1 and 2 belong to the final IMRT plan and result from the third optimization step and not from the decomposition step of our algorithms. In fact, the monitor units have another scale here and are not directly comparable with the delivery time from the segmentation. In contrast, the total change information stems from the decomposition step. Note, that the total change of the segmentation itself is not a significant quantity, because if the matrix entries are large, a larger total change is acceptable. Therefore, we compute the total sum of entries for each intensity matrix and then calculate the relative total change which is the ratio between total change and total sum of entries. The smaller the relative total change, the better is the decomposition.

For the detailed evaluation of our algorithms, we use a set of 264 clinical intensity matrices that originate from electron treatment plans for different patients and beam angles. The matrices are produced during the optimization step 1 of the treatment planning that was introduced in Section 5 and uses the algorithm from [9]. Exemplarily, we compute segmentations for the settings $f = 3$, $g_1 = 3$ and $g_2 = 1$ as well as $f = 4$ and $g_1 = g_2 = 2$. The values of

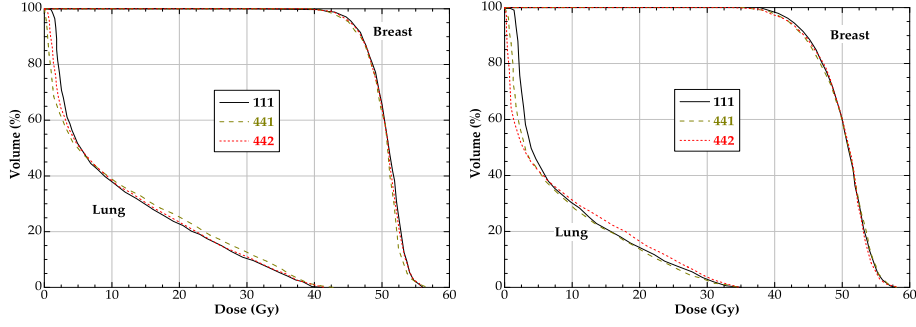


Figure 7: Dose volume histograms for settings 111, 441 and 442 to demonstrate that identical results could be achieved when using greater minimum segment sizes (cf. 111 and 441) and segment shapes without single vertical leaf openings and leaf closings (cf. 441 and 442). The setting xyz is given by a minimum total field height of x , a minimum separation constraint and row overlap of y and a minimum vertical gap of z . Left: Dose volume histogram for optimization setting to reach a better dose coverage of the breast volume. The resulting dose distribution for setting 442 is presented in Figure 6. Right: Dose volume histogram for optimization setting to provide a better sparing of the lung.

the overtravel parameters are also produced in the pre-segmentation step. The results are shown in Table 3 and 4 and demonstrate how much total change is caused respectively avoided by the steps of the algorithms. The overtravel-approximation and MSC-approximation lead to a certain total change of the matrix that is put into the segmentation step. As an exact decomposition in the segmentation step is impossible, the total change increases here again. Both the combination step and the make-two-of-two step try to eliminate segments not satisfying the parameter f and again cause some total change. Finally, the last two steps of the algorithm improve the performance and reduce the approximation error as much as possible. One can see that the combination step and the make-two-of-two-step are performed more often for ASS, as for ASAS the vertical gap ensures that the fields already have a reasonable size after the segmentation step. Of course, the larger the parameters and thus the minimum field size, the larger becomes the total change.

The first column in Table 3 and 4 gives the average results over the 264 matrices, while the second (respectively third) column represent the single results for the matrix with the smallest (respectively largest) relative total change. Homogeneous matrices with large nonzero areas can be decomposed quite well, while matrices with only few nonzero entries that do not span connected areas lead to unacceptable results. As a treatment plan is a superposition of several intensity profiles from different beam angles, the approximation errors balance each other and lead to applicable treatment plans as described above. Furthermore, a certain part of the total change is unavoidable if one requires the constraints (i)-(v), e.g. the total change after MSC-approximation is a good lower bound for the achievable total change. All in all, taking the vertical gap g_2 into account increases the total change while reducing the number of used segments and the monitor units.

Table 1: Segmentation results of IMRT plans using different decomposition settings for optimization setting to achieve the best target coverage. Setting xyz means $f = x$ (minimum total field height), $g_1 = y$ (minimum separation constraint and row overlap) and $g_2 = z$ (minimum vertical gap).

Setting	Mean Relative Total Change	Number of Segments	Number of Monitor Units
111	0.04	90	85586
221	0.16	79	65677
222	0.34	54	63461
331	0.22	49	38598
332	0.36	32	21265
333	0.40	40	36789
441	0.30	55	34262
442	0.41	26	12860
443	0.45	28	13119
444	0.49	27	16899
551	0.35	40	19162
552	0.46	30	10337

Table 2: Segmentation results of IMRT plans using different decomposition settings for optimization setting to reach less dose to the lung. Setting xyz means $f = x$ (minimum total field height), $g_1 = y$ (minimum separation constraint and row overlap) and $g_2 = z$ (minimum vertical gap).

Setting	Mean Relative Total Change	Number of Segments	Number of Monitor Units
111	0.04	78	59399
221	0.16	71	46627
222	0.34	51	42411
331	0.22	52	25896
332	0.36	29	16700
333	0.40	36	24957
441	0.30	45	23509
442	0.41	28	10200
443	0.45	28	9995
444	0.49	27	11318
551	0.35	37	13787
552	0.46	25	8832

7 Conclusion

In the present study, dosimetric and technical constraints have been taken into consideration in intensity-modulated radiation therapy (IMRT). A set of 5 parameters has been introduced, two of them for modelling the leaf overtravel constraint, the other three to ensure a minimum field size and to avoid thin field shapes. We proposed algorithms for approximate segmentation of intensity matrices using segments that satisfy the constraints. We basically distinguish between two approximation problems depending on whether the vertical gap parameter is considered or not. The objective function of the optimization is the deviation between the desired and the approximated intensity profile that has to be minimized. The segmentation step is part of an IMRT optimization process which was examined by comparisons of dose volume histograms of treatment plans with small and large segments as well as with and without thin segment shapes. The histograms show that the use of larger segments results in equal IMRT plans with fewer segments and monitor units respectively. Although the approximation error of the segmentations rises with

Table 3: Numerical results for ASS with parameters $g_1 = 3$ and $f = 3$.

	Average	Rel. TC min.	Rel. TC max.
m	17.55	20	11
n	21.44	19	13
Total sum of entries	886.4	1329	152
Total change	82.24	12	64
Delivery time	15.39	13	3
Number of segments	15	13	3
TC after Overtravel-Approximation	26.34	6	75
TC after MSC-Approximation	57.09	11	88
TC after Segmentation	74.33	13	91
TC change after combination	80.96	28	91
TC after make-two-of-two	93.77	28	101
TC after handle critical segments	88.72	28	94
TC after improvement	82.24	12	94
Combinations	9.19	5	0
Successful make-two-of-two	2.78	0	0
Relative total change	0.16	0.009	0.62

Table 4: Numerical results for ASAS with parameters $g_1 = g_2 = 2$ and $f = 4$.

	Average	Rel. TC min.	Rel. TC max.
m	17.55	23	22
n	21.44	26	28
Total sum of entries	886.4	997	384
Total change	157.7	46	269
Delivery time	7.69	4	4
Number of segments	6.82	3	4
TC after Overtravel-Approximation	26.34	7	89
TC after MSC-Approximation	44.39	7	134
TC after Segmentation	144.8	46	267
TC change after combination	144.8	46	267
TC after make-two-of-two	159.2	46	270
TC after handle critical segments	158.8	46	270
TC after improvement	157.7	46	269
Combinations	0.24	0	0
Successful make-two-of-two	0.57	0	0
Relative total change	0.27	0.05	0.70

increasing minimum field size, equivalent or even better dose distributions could be achieved. Concluding, this first approach to approximated segmentation in IMRT planning shows the potential of these ideas and there is a need for further research in related approximation problems.

References

- [1] T. Achterberg. *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, 2007. <http://opus.kobv.de/tuberlin/volltexte/2007/1611/>.
- [2] R.K. Ahuja and H.W. Hamacher. A network flow algorithm to minimize beam-on time for unconstrained multileaf collimator problems in cancer radiation therapy. *Networks*, 45(1):36–41, 2005.

- [3] D. Baatar, H.W. Hamacher, M. Ehrgott, and G.J. Woeginger. Decomposition of integer matrices and multileaf collimator sequencing. *Discrete Appl. Math.*, 152(1-3):6–34, 2005.
- [4] J.L. Bedford and S. Webb. Constrained segment shapes in direct-aperture optimization for step-and-shoot IMRT. *Med. Phys.*, 33(4):944–958, 2006.
- [5] N. Boland, H. W. Hamacher, and F. Lenzen. Minimizing beam-on time in cancer radiation treatment using multileaf collimators. *Networks*, 43(4):226–240, 2004.
- [6] T.R. Bortfeld, D.L. Kahler, T.J. Waldron, and A.L. Boyer. X-ray field compensation with multileaf collimators. *Int. J. Radiat. Oncol. Biol. Phys.*, 28:723–730, 1994.
- [7] K. Engel. A new algorithm for optimal multileaf collimator field segmentation. *Discrete Appl. Math.*, 152(1-3):35–51, 2005.
- [8] K. Engel and A. Kiesel. Approximated matrix decomposition for IMRT planning with multileaf collimators. *OR Spectrum*, DOI 10.1007/s00291-009-0168-5, 2009.
- [9] K. Engel and E. Tabbert. Fast simultaneous angle, wedge, and beam intensity optimization in inverse radiotherapy planning. *Optimization and Engineering*, 6(4):393–419, 2005.
- [10] C. Engelbeen and S. Fiorini. Constrained decompositions of integer matrices and their applications to intensity modulated radiation therapy. *Networks*, DOI 10.1002/net.20324, 2009.
- [11] C. Engelbeen, S. Fiorini, and A. Kiesel. A closest vector problem arising in radiation therapy planning. *Journal of Combinatorial Optimization*, DOI 10.1007/s10878-010-9308-8, 2010.
- [12] T. Gauer, D. Albers, F. Cremers, R. Harmansa, R. Pellegrini, and R. Schmidt. Design of a computer-controlled multileaf collimator for advanced electron radiotherapy. *Phys. Med. Biol.*, 53:5987–6003, 2006.
- [13] T. Gauer, J. Sokoll, F. Cremers, R. Harmansa, M. Luzzara, and R. Schmidt. Characterization of an add-on multileaf collimator for electron beam therapy. *Phys. Med. Biol.*, 53:1071–1085, 2008.
- [14] T. Kalinowski. A duality based algorithm for multileaf collimator field segmentation with interleaf collision constraint. *Discrete Appl. Math.*, 152(1-3):52–88, 2005.
- [15] T. Kalinowski. Reducing the number of monitor units in multileaf collimator field segmentation. *Phys. Med. Biol.*, 50(6):1147–1161, 2005.
- [16] T. Kalinowski. Multileaf collimator shape matrix decomposition. In *Optimization in Medicine and Biology*. G.J. Lim and E.K.Lee, Auerbach Publishing: 253–286, 2008.
- [17] T. Kalinowski. Reducing the tongue-and-groove underdosage in MLC shape matrix decomposition. *Algorithmic Operations Research*, 3(2), 2008.

- [18] T. Kalinowski. The complexity of minimizing the number of shape matrices subject to minimal beam-on time in multileaf collimator field decomposition with bounded fluence. *Discrete Appl. Math.*, 157:2089–2104, 2009.
- [19] T. Kalinowski. A min cost network flow formulation for approximated MLC segmentation. *Networks*, DOI 10.1002/net.20394, 2010.
- [20] T. Kalinowski and A. Kiesel. Approximated MLC shape matrix decomposition with interleaf collision constraint. *Algorithmic Operations Research*, 4(1):49–57, 2009.
- [21] S. Kamath, S. Sahni, J. Li, J. Palta, and S. Ranka. Leaf sequencing algorithms for segmented multileaf collimation. *Phys. Med. Biol.*, 48(3):307–324, 2003.
- [22] S. Kamath, S. Sahni, J. Palta, S. Ranka, and J. Li. Optimal leaf sequencing with elimination of tongue-and-groove underdosage. *Phys. Med. Biol.*, 49:N7–N19, 2004.
- [23] S. Kamath, S. Sahni, S. Ranka, J. Li, and J. Palta. A comparison of step-and-shoot leaf sequencing algorithms that eliminate tongue-and-groove effects. *Phys. Med. Biol.*, 49:3137–3143, 2004.
- [24] J. Lim, M.C. Ferris, S.J. Wright, D.M. Shepard, and M.A. Earl. An optimization framework for conformal radiation treatment planning. *Inform. Journal on Computing*, 19(3):366–380, 2007.
- [25] M.M. Matuszak, E.W. Larsen, K. Jee, D.L. McShan, and B.A. Fraass. Adaptive diffusion smoothing: A diffusion based method to reduce imrt field complexity. *Med. Phys.*, 35(4):1532–1546, 2008.
- [26] M. Nußbaum. Min cardinality C1 decomposition of integer matrices. Master’s thesis, Faculty for Mathematics, TU Kaiserslautern, 2006.
- [27] W. Que, J. Kung, and J. Dai. ‘Tongue-and-groove’ effect in intensity modulated radiotherapy with static multileaf collimator fields. *Phys. Med. Biol.*, 49:399–405, 2004.
- [28] D.M. Shepard, M.A. Earl, X.A. Li, S. Naqvi, and C. Yu. Direct aperture optimization: A turnkey solution for step-and-shoot IMRT. *Med. Phys.*, 29(6):1007–1018, 2002.
- [29] R. Wunderling. *Paralleleler und objektorientierter Simplex-Algorithmus*. PhD thesis, Technische Universität Berlin, 1996. <http://www.zib.de/Publications/abstracts/TR-96-09/>.

8 Appendix

Algorithm 1 LOC-left

Input: vector v

```

for  $i = \min$  to  $\max$  do
     $tc_{i1} = |v_i - i|$ 
end for
 $tc_{ij} = \infty$  for all  $i \in [\min, \max], j > 1$ 
for  $j = 2$  to  $k$  do
    for  $i = \min$  to  $\max$  do
        for  $i' = \min$  to  $i$  do
            if  $tc_{i,j-1} + |v_j - i'| < tc_{i'j}$  then
                 $tc_{i'j} = tc_{i,j-1} + |v_j - i'|$ 
                 $pre_{i'j} = i$ 
            end if
        end for
    end for
end for
 $opt = \min_{i \in [\min, \max]} tc_{i,k}$ 
Let  $i_{opt}$  be one of the indices with  $tc_{i_{opt},k} = opt$ .
for  $j = k$  down to  $1$  do
     $w_j = i_{opt}$ 
    If still  $j > 1$ , then  $i_{opt} = pre_{i_{opt},j}$ 
end for
Output: vector  $w$ 

```

Algorithm 2 Segmentation ASS

Input: matrix B

```

while  $B \neq 0$  do
     $i = 1$ ;
    while  $b_i = 0$  do
         $l_i = n + 1$ ;  $r_i = 0$ ;  $i = i + 1$ ;
    end while
Choose  $l_i$  and  $r_i$  for row  $i$  such that  $r_i - l_i \geq g_1 - 1$  (see Remark 2).
while  $r_i - l_i \geq g_1 - 1$  and  $b_{i+1} \neq 0$  do
    Find interval ASS: Choose  $l_{i+1}$  and  $r_{i+1}$  for row  $i + 1$  such that the
    overlap with the previous row is  $\geq g_1$  or close row  $i + 1$ .
     $i = i + 1$ ;
end while
Close all remaining rows with  $l_i = n + 1$  and  $r_i = 0$ .
for  $i = 1$  to  $m$  do
    for  $j = l_i$  to  $r_i$  do
         $b_{ij} = b_{ij} - 1$ ;
    end for
end for
Store  $((l_1, r_1), \dots, (l_m, r_m))$  in the segmentation.
end while
Output: Segmentation

```

Algorithm 3 Find interval ASS

Input: i, l_i, r_i

```
 $finish = 0;$   
while  $finish = 0$  do  
   $start = \min\{j \mid b_{i+1,j} > 0\};$   
   $end = \min\{j \mid b_{i+1,j} > b_{i+1,j+1}\};$   
   $l = \max(start, l_i);$   
   $r = \min(end, r_i);$   
  if  $r - l \geq g_1 - 1$  then  
     $l_{i+1} = start; r_{i+1} = end; finish = 1;$   
  else if  $r - l < 0$  then  
     $l_{i+1} = n + 1; r_{i+1} = 0; finish = 1;$   
  else  
    if  $start > l_i$  then  
       $b_{i+1,start-1} = b_{i+1,start-1} + 1;$   
    else  
       $b_{i+1,end+1} = b_{i+1,end+1} + 1;$   
    end if  
  end if  
end while
```

Output: l_{i+1}, r_{i+1}

Remark 2. The interval $[l, r]$ for the first open row is computed analogously to the interval for the other rows, only ignoring the overlap constraint and instead requiring $r - l > g_1 - 1$.

Algorithm 4 Segmentation ASAS

Input: matrix B

```
while  $B \neq 0$  do
   $i = 1; s_{ij} = 0$  for all  $i$  and  $j$ ;
  while  $b_i = 0$  do
     $l_i = n + 1; r_i = 0; i = i + 1$ ;
  end while
   $start = i; closed = 0$ ;
  if  $start > n - g_2 + 1$  then
     $closed = 1$ ;
  else
    Choose  $l_i$  and  $r_i$  for row  $i$  such that  $r_i - l_i \geq g_1 - 1$  or close row  $start$ 
    (see Remark 3).
    if row  $start$  is closed then
       $closed = 1$ ;
    else
      while  $r_i - l_i \geq g_1 - 1$  and  $b_{i+1} \neq 0$  do
        Find interval ASAS: Choose  $l_{i+1}$  and  $r_{i+1}$  for row  $i+1$  such that the
        overlap with the previous row is larger than  $g_1$  or close row  $i + 1$ .
         $i = i + 1$ ;
      end while
      for all remaining rows  $i$  do
        if  $s_i \neq 0$  then
           $l_i = \text{index of the first one in } s_i; r_i = \text{index of the last one in } s_i$ ;
        else
           $l_i = n + 1, r_i = 0$ ;
        end if
      end for
    end if
  end if
  if  $closed = 1$  then
     $b_{start} = 0$ ;
  else
    for  $i = 1$  to  $m$  do
      for  $j = l_i$  to  $r_i$  do
        if  $b_{ij} > 0$  then
           $b_{ij} = b_{ij} - 1$ ;
        end if
      end for
    end for
    Store  $((l_1, r_1), \dots, (l_m, r_m))$  in the segmentation.
  end if
end while
```

Output: Segmentation

Algorithm 5 Find interval ASAS

Input: $i + 1, l_i, r_i$, current matrix S

if $s_i = 0$ **then**

$v_1 = n + 1; v_2 = 0;$

else

$v_1 = \text{index of the first one in } s_i; v_2 = \text{index of the last one in } s_i;$

end if

$t_1 = \text{index of the first positive entry in } b_i;$

$t_2 = \text{index of the last positive entry in } b_i;$

$opt = 0; l_{i+1} = n + 1; r_{i+1} = 0;$

for $l = \min(v_1, t_1)$ **to** $\min(b_l, v_1)$ **do**

for $r = \max(l + g_1 - 1, v_2, b_r)$ **to** $\max(v_2, t_2)$ **do**

if $\min(r, r_i) - \max(l, l_i) \geq g_1 - 1$ **and not** $(l < l_1 \text{ and } i + 1 > m - g_2 + 1)$
 and not $(r > r_i \text{ and } i + 1 > m - g_2 + 1)$ **and no unavoidable zero is**
 between } l \text{ and } r **then**

$benchmark = 0;$

for all entries (k, j) **of unavoidable ones with** $i + 2 \leq k \leq i + g_2$ **do**

if $b_{kj} > 0$ **then**

$benchmark = benchmark - 1;$

else

$benchmark = benchmark + 1;$

end if

end for

for $j = l$ **to** r **do**

if $b_{i+1,j} > 0$ **then**

$benchmark = benchmark - 1;$

else

$benchmark = benchmark + 1;$

end if

end for

if $benchmark < opt$ **then**

$opt = benchmark; l_{i+1} = l; r_{i+1} = r;$

end if

end if

end for

end for

 Put unavoidable ones or zeros corresponding to the decision for l_{i+1} and r_{i+1} into S .

Output: l_{i+1}, r_{i+1} , current matrix S

Remark 3. The choice of (l, r) in row *start* is again computed analogously, only ignoring the overlap constraint with the previous row.

Algorithm 6 Total Change Improvement

Input: Segmentation**for** all segments **do****for** $i = 1$ to m **do****if** $l_i > 1$ and $b_{i,l_i-1} < a_{i,l_i-1}$ (for ASAS: and decreasing l_i will not violate (iv)) **then** $l_i = l_i - 1;$ **end if****if** $r_i < 1$ and $b_{i,r_i+1} < a_{i,r_i+1}$ (for ASAS: and increasing r_i will not violate (iv)) **then** $r_i = r_i + 1;$ **end if****if** $b_{i,l_i} > a_{i,l_i}$ and increasing l_i will not violate (iii)(for ASAS: and increasing l_i will not violate (iv)) **then** $l_i = l_i + 1;$ **end if****if** $b_{i,r_i} > a_{i,r_i}$ and decreasing r_i will not violate (iii)(for ASAS: and decreasing r_i will not violate (iv)) **then** $r_i = r_i - 1;$ **end if****end for**

Update approximation matrix

end for**Output:** Segmentation
